ARTICLE IN PRESS

Available online at www.sciencedirect.com

ScienceDirect

Advances in Space Research xxx (2014) xxx–xxx

ADVANCES IN SPACE RESEARCH
(a COSPAR publication)

www.elsevier.com/locate/asr

A comparison of iterative explicit guidance algorithms for space launch vehicles

Eun-Jung Song [*], Sangbum Cho, Woong-Rae Roh

Korea Aerospace Research Institute, 169-84 Gwahak-ro, Yuseong-gu, Daejeon 305-806, Republic of Korea

Received 5 February 2014; received in revised form 14 August 2014; accepted 18 September 2014

Abstract

This paper analyzes and compares the performance of two prominent explicit guidance algorithms: the iterative guidance mode and the powered explicit guidance. We performed a series of numerical simulations of a space launch vehicle model for both nominal and off-nominal conditions. One of our findings is that if we take into account the originally ignored higher-order terms from the guidance parameters of the iterative guidance mode for a long-range flight, the guidance performance can be enhanced to a level comparable to that of the powered explicit guidance. These higher-order terms can be included by employing an iterative predictor–corrector method like the powered explicit guidance. Also we proposed a remedy of preventing relatively earlier divergence of the guidance commands with the predictor–corrector iteration than that of the linear differential corrector approach by making a better initial guess.

Keywords: Explicit guidance law; Space launch vehicle; Thrust acceleration integrals; Iterative methods; Guidance performance analysis

1. Introduction

Most of the modern multi-staged space launch vehicles adopt closed-loop guidance laws for the exoatmospheric trajectory, although open-loop ascent guidance is effective enough for the first stage maneuver against the atmospheric loads. One of the main virtues of a closed-loop guidance program is that the steering vectors and engine cutoff times can be determined to achieve the target with minimum fuel consumption. For this purpose, various explicit guidance schemes based on linear tangent laws have been developed since the space era of 1960s and their inherent mission flexibility has allowed them to be applied even in today's space launch vehicles (Shrivastava et al., 1986). To handle more complicated mission requirements for future space transportation, guidance algorithms with higher degree of onboard adaptability have been being developed (Hardtla et al., 1987; Jessick and Knobbs, 1992), including those based on onboard trajectory optimization with advances in computer processors (Skalecki and Martin, 1993; Jutty et al., 2000). Other research efforts, also principally indebted to much improved onboard computer capabilities, have been devoted to eliminate simplifying assumptions previously required to derive the closed-form solutions of guidance problems (Delporte and Sauvient, 1992).

It could be said that the iterative guidance mode (IGM) developed for guidance of the Saturn launch vehicles (Chandler and Smith, 1967) and the powered explicit guidance (PEG) algorithm developed during the Space Shuttle Program with various missions (McHenry et al., 1979; Schleich, 1982) are among the most distinguished and flight-proven guidance algorithms ever devised. Although the performance of these algorithms has been demonstrated by hundreds of successful flights, it is rather hard to find open literatures analyzing the effects of the approximations used in their derivation or comparing the

[*] Corresponding author. Tel.: +82 42 860 2994.
E-mail address: ejsong@kari.re.kr (E.-J. Song).

http://dx.doi.org/10.1016/j.asr.2014.09.025
0273-1177/© 2014 COSPAR. Published by Elsevier Ltd. All rights reserved.

performance of these two algorithms. Song et al. (2011) proposed a method to reduce the prediction errors of IGM by taking into account the effect of approximate formula of the angle-to-go prediction to orbit injection accuracy. A higher-order form of PEG was proposed by Sinha and Shrivastava (1990) to consider the high maneuvers of Indian launch vehicles, and they showed that its performance was superior to that of the original form. Vittal and Bhat (1991) and Delporte and Sauvient (1992) formulated the thrust acceleration integrals with constant thrust analytically for the vector form of the linear tangent law, which had been originally approximated by Taylor series values (McHenry et al., 1979; Sinha and Shrivastava, 1990).

This study analyzes and compares the performances of IGM and PEG while seeking methods to improve the performance of these guidance schemes in their original forms. We investigated the effect of higher-order nonlinear terms previously ignored in IGM to derive the approximate linear equations of guidance parameters by employing similar approaches by Sinha and Shrivastava (1990) with PEG algorithm. That is, higher-order terms are shown to be included in almost the same way with PEG algorithm if an iterative predictor–corrector method is adopted. There are some references commenting that PEG can be considered as a kind of a vector form of IGM and the performance of these two algorithms is nearly identical for vacuum flight phase (Jaggers, 1977; Hanson et al., 1995). However, simulation results show that the performance of IGM can be more deteriorated than that of PEG, especially in terms of slower convergence in the time-to-go prediction. It is shown that the performance of IGM can be improved by introducing previously ignored higher-order terms, which again makes the linear angle command assumption of the algorithm more valid.

The guidance parameters of PEG are computed using the iterative predictor–corrector approach starting with an initial guess at every major guidance computation cycle. By comparing with the linear differential correction method using first derivatives to all guidance parameters (Delporte and Sauvient, 1992), it is shown that the initial guess of the predictor–corrector approach is more erroneous as the time-to-go approaches 0, which may lead to more iterations. It is well known that, for iterative algorithms, the first guess close to the answer significantly improves the convergence properties (Luenberger and Ye, 2008). The same applies to the initial guess of the velocity-to-go, the running variable of the iteration process in PEG. In this study, we showed that the convergence of the iteration process in PEG can be enhanced by improving the initial guess of the velocity-to-go.

Although our approaches here may not claim significant performance improvement on these well-developed schemes in their original form, we proved that some deficiencies can be overcome without much more complexities such as onboard optimization methods which are currently one of the major research topics in the space launch vehicle guidance area. Except for this introduction, this paper is organized into four sections. The formulae of higher-order form of IGM are elaborated first and then described is a comparison between the predictor–corrector approach of PEG and the linear differential corrector method. Following the simulation results for the proposed approaches, the conclusions of this work are summarized.

2. Explicit guidance law with higher-order nonlinear terms

Most of the explicit guidance laws have a functional form based on an optimal trajectory minimizing the usage of propellants. Derivation of IGM is also based on the fact that the optimal thrust direction for the flat-earth and uniform gravity model is approximately given by a linear angle form (Chandler and Smith, 1967). The pitch and yaw attitude commands are assumed to be linear relative to the guidance reference system G as follows, with G being defined in the target orbit plane whose origin is located at the center of the Earth.

$$\theta_G = \theta_{G_v} - \left(\theta_{Gp} - \dot{\theta}_G t\right) \tag{1}$$

$$\psi_G = \psi_{G_v} - \left(\psi_{Gp} - \dot{\psi}_G t\right) \tag{2}$$

where current time is set to 0, and t denotes the time varying from 0 to time-to-go, t_{go}, the remaining time before achieving the target injection point. The x-axis of the guidance coordinate system G goes toward the predicted orbit injection point from the origin, the y-axis lies in the normal direction of the target orbit plane, and the z-axis completes the right-handed coordinate system. Among the guidance parameters, $\theta_{G_v}, \theta_{G_p}, \dot{\theta}_G, \psi_{G_v}, \psi_{G_p}$, and $\dot{\psi}_G$ are related to the commanded attitudes (thrust direction) and t_{go} determines the time of the engine shutdown. Finally range-angle-to-go, ξ_{go}, an angle between current position and predicted target injection point, should be computed to define the G frame. These parameters are calculated to satisfy the position and velocity constraints of a target orbit by analytic integration of the following point-mass equations of motion applicable to a typical space launch vehicle in a vacuum.

$$\ddot{x}^G = a \sin \theta_G \cos \psi_G + g_x^G \tag{3}$$

$$\ddot{y}^G = a \sin \psi_G + g_y^G \tag{4}$$

$$\ddot{z}^G = a \cos \theta_G \cos \psi_G + g_z^G \tag{5}$$

where the first terms represent the acceleration by thrust and the second terms by gravity, which are integrated independently.

The derivation procedure of the guidance parameters for IGM in its original (Chandler and Smith, 1967) and a modified form to consider previously ignored higher-order terms, is as follows. First, θ_{G_v} and ψ_{G_v} are derived to satisfy only the terminal velocity constraints under the assumption of invariant thrust direction. Integration of Eqs. (3)–(5) during the remaining burn time leads to

$$\dot{x}_f^G = \dot{x}_o^G + \sin\theta_{G_v}\ \cos\psi_{G_v}\int_0^{t_{go}} adt + v_{grav_x}^G \tag{6}$$

$$\dot{y}_f^G = \dot{y}_o^G + \sin\psi_{G_v}\int_0^{t_{go}} adt + v_{grav_y}^G \tag{7}$$

$$\dot{z}_f^G = \dot{z}_o^G + \cos\theta_{G_v}\cos\psi_{G_v}\int_0^{t_{go}} adt + v_{grav_z}^G. \tag{8}$$

Then the guidance parameters are obtained as follows.

$$\theta_{G_v} = \tan^{-1}\left(\frac{\Delta v_x^G}{\Delta v_z^G}\right), \quad \psi_{G_v} = \tan^{-1}\left(\frac{\Delta v_y^G}{\sqrt{\Delta v_x^{G^2} + \Delta v_z^{G^2}}}\right) \tag{9}$$

where

$$\Delta\vec{v}^G = \dot{\vec{v}}_f^G - \dot{\vec{v}}_o^G - \dot{\vec{v}}_{grav}^G \tag{10}$$

Here t_{go} is derived from the following relationship.

$$\left|\Delta\vec{v}^G\right| = \int_0^{t_{go}} adt \tag{11}$$

Then the remaining attitude guidance parameters $\theta_{G_p}, \dot{\theta}_G, \psi_{G_p}$ and $\dot{\psi}_G$ are derived to satisfy the terminal position constraints in the directions of x and y axes. To derive analytic expressions for the thrust integrals, $(\theta_{G_p} - \dot{\theta}_G t)$ and $(\psi_{G_p} - \dot{\psi}_G t)$ are assumed to be small enough so that the sine and cosine functions of t can be substituted by series approximation as follows.

$$\sin(\theta_{G_p} - \dot{\theta}_G t) \approx \underline{(\theta_{G_p} - \dot{\theta}_G t)} - \frac{\left(\theta_{G_p} - \dot{\theta}_G t\right)^3}{3!} + \cdots \tag{12}$$

$$\cos\left(\theta_{G_p} - \dot{\theta}_G t\right) \approx \underline{1} - \underline{\underline{\frac{(\theta_{G_p} - \dot{\theta}_G t)^2}{2!}}} + \frac{\left(\theta_{G_p} - \dot{\theta}_G t\right)^4}{4!} + \cdots \tag{13}$$

$$\sin\left(\psi_{G_p} - \dot{\psi}_G t\right) \approx \underline{(\psi_{G_p} - \dot{\psi}_G t)} - \frac{\left(\psi_{G_p} - \dot{\psi}_G t\right)^3}{3!} + \cdots \tag{14}$$

$$\cos\left(\psi_{G_p} - \dot{\psi}_G t\right) \approx \underline{1} - \underline{\underline{\frac{\left(\psi_{G_p} - \dot{\psi}_G t\right)^2}{2!}}} + \frac{\left(\psi_{G_p} - \dot{\psi}_G t\right)^4}{4!} + \cdots \tag{15}$$

The original IGM algorithm considers only the first underlined terms in Eqs. (12)–(15) and such ignorance of higher-order terms is valid only when time-to-go is sufficiently small. From these disregarded terms, we can infer that, for long range cases, the thrust integrals may not be accurate and even deteriorate the guidance performance. According to the simplification adopted in the original IGM, the x-axis component of thrust acceleration in Eq. (3) can be obtained as follows.

$$\begin{aligned}\sin\theta_G &= \sin\left(\theta_{G_v} - \left(\theta_{G_p} - \dot{\theta}_G t\right)\right)\\ &\approx \sin\theta_{G_v} - \left(\theta_{G_p} - \dot{\theta}_G t\right)\cos\theta_{G_v}\end{aligned} \tag{16}$$

$$\begin{aligned}\cos\psi_G &= \cos\left(\psi_{G_v} - \left(\psi_{G_p} - \dot{\psi}_G t\right)\right)\\ &\approx \cos\psi_{G_v} + \left(\psi_{G_p} - \dot{\psi}_G t\right)\sin\psi_{G_v}\end{aligned} \tag{17}$$

$$\begin{aligned}\sin\theta_G\cos\psi_G =& \sin\theta_{G_v}\cos\psi_{G_v} - \left(\theta_{G_p} - \dot{\theta}_G t\right)\cos\theta_{G_v}\cos\psi_{G_v}\\ &+ \left(\psi_{G_p} - \dot{\psi}_G t\right)\sin\theta_{G_v}\sin\psi_{G_v}\\ &- \left(\theta_{G_p} - \dot{\theta}_G t\right)\left(\psi_{G_p} - \dot{\psi}_G t\right)\cos\theta_{G_v}\sin\psi_{G_v}\end{aligned} \tag{18}$$

It can be noticed that the original IGM does not consider all the second-order terms in an exact manner. From Eqs. (16)–(18) it may appear that all terms less than or equal to the second-order terms of t are considered, however, if the terms with double underlines in Eqs. (13) and (15) are not neglected from the first step of the algorithm derivation, then as in Eq. (19), another second-order term should show which may have equal importance to other second-order terms in Eq. (18). Likewise the third- and more higher-order terms also need to be included as shown in Eq. (19) for accuracy enhancement.

$$\begin{aligned}\sin\theta_G\cos\psi_G =& \underline{\sin\theta_{G_v}\ \cos\psi_{G_v} - \left(\theta_{G_p} - \dot{\theta}_G t\right)\cos\theta_{G_v}\cos\psi_{G_v} + \left(\psi_{G_p} - \dot{\psi}_G t\right)\sin\theta_{G_v}\sin\psi_{G_v}}\\ &\underline{-\left(\theta_{G_p} - \dot{\theta}_G t\right)\left(\psi_{G_p} - \dot{\psi}_G t\right)\cos\theta_{G_v}\sin\psi_{G_v}}\\ &\underline{\underline{-\frac{1}{2}\left(\theta_{G_p} - \dot{\theta}_G t\right)^2\sin\theta_{G_v}\cos\psi_{G_v} - \frac{1}{2}\left(\psi_{G_p} - \dot{\psi}_G t\right)^2\sin\theta_{G_v}\cos\psi_{G_v}}}\\ &+\frac{1}{4}\left(\theta_{G_p} - \dot{\theta}_G t\right)^2\left(\psi_{G_p} - \dot{\psi}_G t\right)^2\sin\theta_{G_v}\cos\psi_{G_v}\cdots\end{aligned} \tag{19}$$

Using Eq. (19), the x-axis velocity component is obtained by integration of Eq. (3) during t_{go} and additional integration gives the x-axis position component as follows.

$$\begin{aligned}\dot{x}_f^G =& \dot{x}_o^G + \int_0^{t_{go}} a\sin\theta_G\cos\psi_G dt + v_{grav_x}^G\\ =& \dot{x}_o^G + \underline{L\sin\theta_{G_v}\ \cos\psi_{G_v} - \left(L\theta_{G_p} - J\dot{\theta}_G\right)\cos\theta_{G_v}\cos\psi_{G_v} + \left(L\psi_{G_p} - J\dot{\psi}_G\right)\sin\theta_{G_v}\sin\psi_{G_v}}\\ &\underline{-\theta_{G_p}\left(L\psi_{G_p} - J\dot{\psi}_G\right)\cos\theta_{G_v}\sin\psi_{G_v} + \dot{\theta}_G\left(J\psi_{G_p} - H\dot{\psi}_G\right)\cos\theta_{G_v}\sin\psi_{G_v}}\\ &\underline{\underline{-\frac{1}{2}\left(L\theta_{G_P}^2 - 2J\theta_{G_p}\dot{\theta}_G + H\dot{\theta}_G^2\right)\sin\theta_{G_v}\cos\psi_{G_v} - \frac{1}{2}\left(L\psi_{G_P}^2 - 2J\psi_{G_p}\dot{\psi}_G + H\dot{\psi}_G^2\right)\sin\theta_{G_v}\ \cos\psi_{G_v}}}\\ &+\frac{1}{4}\sin\theta_{G_v}\ \cos\psi_{G_v}\{L\theta_{G_P}^2\psi_{G_P}^2 - 2J\left(\theta_{G_P}^2\psi_{G_p}\dot{\psi}_G + \theta_{G_p}\dot{\theta}_G\psi_{G_P}^2\right)\cdots + v_{grav_x}^G\end{aligned} \tag{20}$$

$$\begin{aligned}x_f^G =& x_o^G + \dot{x}_o^G t_{go} + \int_0^{t_{go}}\int_0^t a\sin\theta_G\cos\psi_G dsdt + p_{grav_x}^G\\ =& x_o^G + \dot{x}_o^G t_{go} + \underline{S\sin\theta_{G_v}\ \cos\psi_{G_v} - \left(S\theta_{G_p} - Q\dot{\theta}_G\right)\cos\theta_{G_v}\ \cos\psi_{G_v} + \left(S\psi_{G_p} - \dot{\psi}_G Q\right)\sin\theta_{G_v}\ \sin\psi_{G_v}}\\ &-\theta_{G_p}\left(S\psi_{G_p} - Q\dot{\psi}_G\right)\cos\theta_{G_v}\ \sin\psi_{G_v} + \dot{\theta}_G\left(Q\psi_{G_p} - P\dot{\psi}_G\right)\cos\theta_{G_v}\ \sin\psi_{G_v}\\ &\underline{-\frac{1}{2}\left(S\theta_{G_P}^2 - 2Q\theta_{G_p}\dot{\theta}_G + P\dot{\theta}_G^2\right)\sin\theta_{G_v}\cos\psi_{G_v} - \frac{1}{2}\left(S\psi_{G_P}^2 - 2Q\psi_{G_p}\dot{\psi}_G + P\dot{\psi}_G^2\right)\sin\theta_{G_v}\cos\psi_{G_v}}\\ &+\frac{1}{4}\sin\theta_{G_v}\cos\psi_{G_v}\{S\theta_{G_P}^2\psi_{G_P}^2 - 2Q\left(\theta_{G_P}^2\psi_{G_p}\dot{\psi}_G + \theta_{G_p}\dot{\theta}_G\psi_{G_P}^2\right)\cdots + p_{grav_x}^G\end{aligned} \tag{21}$$

where L, J, H, S, Q, and P are defined as follows.

$$\begin{aligned}L &= \textstyle\int_o^{t_{go}} adt \qquad & S &= \textstyle\int_o^{t_{go}}\int_o^t adsdt\\ J &= \textstyle\int_o^{t_{go}} atdt \qquad & Q &= \textstyle\int_o^{t_{go}}\int_o^t asdsdt\\ H &= \textstyle\int_o^{t_{go}} at^2dt \qquad & P &= \textstyle\int_o^{t_{go}}\int_o^t as^2dsdt\end{aligned} \tag{22}$$

For analytic integration of Eq. (22) over the burning time, constant thrust or constant acceleration models are usually employed. In a similar manner, for the y-axis velocity and position components,

Please cite this article in press as: Song, E.-J., et al. A comparison of iterative explicit guidance algorithms for space launch vehicles. J. Adv. Space Res. (2014), http://dx.doi.org/10.1016/j.asr.2014.09.025

$$\begin{aligned}\dot{y}_f^G &= \dot{y}_o^G + \int_0^{t_{go}} a\sin\psi_G dt + v_{grav_y}^G \\ &= \dot{y}_o^G + \underline{L\sin\psi_{G_v} - \left(L\psi_{G_p} - J\dot{\psi}_G\right)\cos\psi_{G_v}} \\ &\quad \underline{\underline{-\frac{1}{2}\left(L\psi_{G_P}^2 - 2J\psi_{G_p}\dot{\psi}_G + H\dot{\psi}_G^2\right)\sin\psi_{G_v}}}\cdots + v_{grav_y}^G \end{aligned} \quad (23)$$

$$\begin{aligned}y_f^G &= y_o^G + \dot{y}_o^G t_{go} + \int_0^{t_{go}}\int_0^{t} a\sin\psi_G dsdt + p_{grav_y}^G \\ &= y_o^G + \dot{y}_o^G t_{go} + \underline{S\sin\psi_{G_v} - \left(S\psi_{G_p} - Q\dot{\psi}_G\right)\cos\psi_{G_v}} \\ &\quad \underline{\underline{-\frac{1}{2}\left(S\psi_{G_P}^2 - 2Q\psi_{G_P}\dot{\psi}_G + P\dot{\psi}_G^2\right)\sin\psi_{G_v}}}\cdots + p_{grav_y}^G \end{aligned} \quad (24)$$

Many real mission designers have taken the great possible care to avoid large yaw maneuvers to change the orientation of the orbital plane since these kinds of maneuvers are not economical in terms of propellant consumption. In that case the higher-order terms of $(\psi_{G_p} - \dot{\psi}_G t)$ in the yaw motion can be neglected. Then the four equations, Eqs. (20), (21), (23) and (24), are used to compute the four unknowns $\theta_{G_p}, \dot{\theta}_G, \psi_{G_p}$, and $\dot{\psi}_G$, and the velocity equations of Eqs. (20) and (23) come to the same equations as Eqs. (6) and (7), respectively, from the following equations.

$$\begin{aligned}&\underline{-\left(L\theta_{G_p} - J\dot{\theta}_G\right)\cos\theta_{G_v}\cos\psi_{G_v} + \left(L\psi_{G_p} - J\dot{\psi}_G\right)\sin\theta_{G_v}\sin\psi_{G_v}} \\ &\quad \underline{-\theta_{G_p}\left(L\psi_{G_p} - J\dot{\psi}_G\right)\cos\theta_{G_v}\sin\psi_{G_v} + \dot{\theta}_G\left(J\psi_{G_p} - H\dot{\psi}_G\right)\cos\theta_{G_v}\sin\psi_{G_v}} \\ &\quad \underline{\underline{-\frac{1}{2}\left(L\theta_{G_P}^2 - 2J\theta_{G_p}\dot{\theta}_G + H\dot{\theta}_G^2\right)\sin\theta_{G_v}\cos\psi_{G_v} - \frac{1}{2}\left(L\psi_{G_P}^2 - 2J\psi_{G_p}\dot{\psi}_G + H\dot{\psi}_G^2\right)\sin\theta_{G_v}\cos\psi_{G_v}}} \\ &\quad + \frac{1}{4}\sin\theta_{G_v}\cos\psi_{G_v}\{L\theta_{G_P}^2\psi_{G_P}^2 - 2J\left(\theta_{G_P}^2\psi_{G_p}\dot{\psi}_G + \theta_{G_p}\dot{\theta}_G\psi_{G_P}^2\right)\cdots = 0 \end{aligned} \quad (25)$$

$$\underline{-\left(L\psi_{G_p} - J\dot{\psi}_G\right)\cos\psi_{G_v}}\ \underline{\underline{-\frac{1}{2}\left(L\psi_{G_P}^2 - 2J\psi_{G_p}\dot{\psi}_G + H\dot{\psi}_G^2\right)\sin\psi_{G_v}}}\cdots = 0 \quad (26)$$

For the velocity in z-axis direction,

$$\begin{aligned}\dot{z}_f^G &= \dot{z}_o^G + \int_0^{t_{\mathrm{go}}} a\cos\theta_G\cos\psi_G \mathrm{dt} + v_{\mathrm{grav}_z}^G \\ &= \dot{z}_o^G + \underline{L\cos\theta_{G_v}\cos\psi_{G_v} + \left(L\theta_{G_p} - J\dot{\theta}_G\right)\sin\theta_{G_v}\cos\psi_{G_v} + \left(L\psi_{G_p} - J\dot{\psi}_G\right)\cos\theta_{G_v}\sin\psi_{G_v}} \\ &\quad \underline{+\theta_{G_p}\left(L\psi_{G_p} - J\dot{\psi}_G\right)\sin\theta_{G_v}\sin\psi_{G_v} - \dot{\theta}_G\left(J\psi_{G_p} - H\dot{\psi}_G\right)\sin\theta_{G_v}\sin\psi_{G_v}} \\ &\quad \underline{-\frac{1}{2}\left(L\theta_{G_P}^2 - 2J\theta_{G_p}\dot{\theta}_G + H\dot{\theta}_G^2\right)\cos\theta_{G_v}\cos\psi_{G_v}} \\ &\quad \underline{\underline{-\frac{1}{2}\left(L\psi_{G_P}^2 - 2J\psi_{G_p}\dot{\psi}_G + H\dot{\psi}_G^2\right)\cos\theta_{G_v}\cos\psi_{G_v}}} \\ &\quad + \frac{1}{4}\cos\theta_{G_v}\cos\psi_{G_v}\{L\theta_{G_P}^2\psi_{G_P}^2 - 2J\left(\theta_{G_P}^2\psi_{G_p}\dot{\psi}_G + \theta_{G_p}\dot{\theta}_G\psi_{G_P}^2\right)\cdots + v_{\mathrm{grav}_z}^G \end{aligned} \quad (27)$$

For the original IGM, approximate linear equations are derived for the four parameters $\theta_{G_p}, \dot{\theta}_G, \psi_{G_p}$, and $\dot{\psi}_G$ by ignoring the higher order terms. In that case, Eq. (27) also becomes the same as Eq. (8). The z-axis position equation can be derived in a similar manner, where the constraint value is 0 from the definition of the G frame. The guidance parameter, ξ_{go} can be computed from the z-axis position equation, however, Chandler and Smith (1967) introduced a mission-dependent constant K_g to calculate ξ_{go} approximately to avoid solving the z-axis position equation, as follows.

$$\xi_{go} = \frac{1}{r_f}\left(vt_{go} + S + K_g t_{go}^2\right) \quad (28)$$

where K_g is calculated before each flight through the computer simulations of the flight trajectories.

The guidance algorithm is a procedure to solve the equations for the guidance parameters. The unknown parameters are $\theta_{G_v}, \theta_{G_p}, \dot{\theta}_G, \psi_{G_v}, \psi_{G_p}, \dot{\psi}_G, t_{go}$, and ξ_{go} (8 unknowns), and they must satisfy the target position and velocity constraints (6 equations) and two additional velocity constraints (2 equations) to simplify the procedure of finding solutions by formulating the approximate linear equations (Chandler and Smith, 1967). In that case, with given t_{go}, the iteration is performed only for t_{go} and ξ_{go}. Then the guidance equations consisting of only the first-order and some selected second-order terms of the guidance parameters related to maneuvers can be solved analytically.

On the other hand, in order to consider the higher-order nonlinear terms, the iteration must include the maneuver-related guidance parameters, and thus is accompanied by a little more computational complexity. To solve nonlinear equations with higher-order terms, either the predictor–corrector approach or the linear differential method can be employed. Between these two of the most prevailing solvers, we chose the former approach here as in the original PEG. A predictor–corrector algorithm consists of two separate steps to find a solution: the prediction step to calculate an approximation of the desired quantity, and then the correction step to refine the initial approximation using a different method. Likewise, in our approach, the initial guidance parameters are computed by solving the approximate linear equations as in the case of the original IGM and then the higher-order terms are computed using the approximate solution. In the next iteration step, the higher-order terms ignored are added to the linear equations as constant bias terms. In this way, the solutions can be improved as the iteration proceeds. A major advantage of the original IGM is that it does not require massive iterations, at the same time it causes limited upper bounds for the applicable region (Karacsony and Cole, 1970). By including the higher-order terms as proposed in this work, the applicability of IGM can be easily extended and the corresponding computational cost is expected to be similar to that of PEG.

3. Explicit guidance law using predictor–corrector algorithm

The guidance parameters of PEG are slightly different from those of IGM, since the thrust direction is given by a simplified vector form of the linear tangent steering law as shown in Eq. (29), which is derived from an ascent trajectory solution with minimum fuel consumption (McHenry et al., 1979).

$$\vec{u}_F = \frac{\vec{\lambda}_v + \dot{\vec{\lambda}}(t - t_\lambda)}{\sqrt{1 + \dot{\lambda}^2 (t - t_\lambda)^2}} \approx \vec{\lambda}_v \{1 - \frac{1}{2}\dot{\lambda}^2 (t - t_\lambda)^2\} + \dot{\vec{\lambda}}(t - t_\lambda) \tag{29}$$

where

$$|\vec{\lambda}_v| = 1, \quad \vec{\lambda}_v \cdot \dot{\vec{\lambda}} = 0 \tag{30}$$

As briefly mentioned in the previous section, the predictor–corrector computational sequence is adopted to solve the guidance parameters satisfying the cutoff constraints in PEG algorithm. The seven unknown parameters $\vec{\lambda}_v, \dot{\vec{\lambda}}$, and t_{go} are computed in iterative fashion using $\vec{v}_{go}$ as a running variable. The velocity-to-go of each major loop is calculated to be diminished by the velocity change $\Delta\vec{v}$ from the vehicle velocity of the previous iteration cycle.

$$\vec{v}_{go}(n) = \vec{v}_{go}(n-1) - \Delta\vec{v} \tag{31}$$

With given $\vec{v}_{go}$, the prediction process starts with evaluation of t_{go} and the thrust integrals defined in Eq. (22). Then the first and second integrations of the point-mass equations of motion using the thrust direction form in Eq. (29) give the following linear relationships among the guidance parameters and cutoff conditions.

$$\vec{\lambda}_v = \frac{\vec{v}_{go}}{L} \tag{32}$$

$$\dot{\vec{\lambda}} = \frac{\vec{r}_{go} - S\vec{\lambda}_v}{(Q - St_\lambda)} \tag{33}$$

$$t_\lambda = \frac{J}{L} \tag{34}$$

where

$$\vec{r}_{go} = \vec{r}_d - \vec{r} - \vec{v}t_{go} - \vec{r}_{grav} + \vec{r}_{bias} \tag{35}$$

The bias term, $\vec{r}_{bias} = \vec{r}_{go} - \vec{r}_{thrust} \approx \frac{1}{2}\dot{\lambda}^2 (P - 2Qt_\lambda + St_\lambda^2)$ $\vec{\lambda}_v$, is computed using the guidance parameters obtained from the previous iteration step. In this way, the second- and higher-order nonlinear terms can be included as a bias term and considered in the prediction of the cutoff velocity and position.

$$\vec{v}_p = \vec{v} + \vec{v}_{thrust} + \vec{v}_{grav} \tag{36}$$

$$\vec{r}_p = \vec{r} + \vec{v}t_{go} + \vec{r}_{thrust} + \vec{r}_{grav} \tag{37}$$

where

$$\vec{v}_{thrust} \approx \vec{\lambda}_v \left\{ L - \frac{1}{2}\dot{\lambda}^2 (H - Jt_\lambda) \right\} \tag{38}$$

$$\vec{r}_{thrust} \approx \vec{\lambda}_v \left\{ S - \frac{1}{2}\dot{\lambda}^2 \left(P - 2Qt_\lambda + St_\lambda^2\right) \right\} + (Q - St_\lambda)\dot{\vec{\lambda}}. \tag{39}$$

Next in the correction process, the desired cutoff condition and the velocity-to-go are computed using the predicted states and the target orbit information. Convergence can be determined based on the updated time-to-go not changing significantly from that of the previous iteration step.

The foregoing seven guidance parameters $\vec{\lambda}_v, \dot{\vec{\lambda}}$, and t_{go} of PEG can also be computed using the linear differential correction method as proposed by Delporte and Sauvient (1992). The nonlinear relationship between the guidance parameters and the desired cutoff constraints can be represented as follows.

$$Y = f(U) \tag{40}$$

where

$$U = \begin{bmatrix} \vec{\lambda}_v \\ \dot{\vec{\lambda}} \\ t_{go} \end{bmatrix}, \quad Y^{\text{target}} = \begin{bmatrix} v^{\text{target}} \\ \gamma^{\text{target}} \\ r^{\text{target}} \\ i^{\text{target}} \\ \Omega^{\text{target}} \\ 1 \\ 0 \end{bmatrix} \tag{41}$$

The former five constraints from Eq. (41) represent the desired position and velocity, which are usually given by target orbital parameters, and the latter two constraints are derived from Eq. (30). The guidance parameters U are updated until the targeting error ΔY becomes close to 0.

$$\Delta Y = \frac{\partial f}{\partial U}\Delta U \tag{42}$$

$$U \leftarrow U - \Delta U \tag{43}$$

The difference between the predicted end points and the targets, ΔY, can be computed by analytical integration of the flight trajectory in the same way as for PEG starting with an initial guess of U given from the previous guidance cycle. The sensitivity matrix, $\frac{\partial f}{\partial U}$, can also be evaluated analytically from the nonlinear relationship between U and Y. Detailed description of the derivatives of the constraints with respect to the guidance parameters are presented in Delporte and Sauvient (1992).

Song et al. (2013) compared the predictor–corrector approach of PEG with the linear differential corrector method and showed that the guidance parameters obtained from both methods became very similar if the thrust and gravity acceleration integrals needed in trajectory integrations were computed in the same way. However their iterative algorithms described above imply that, regardless of the similarity between the converged solutions, intermediate solutions from the iteration processes can be slightly different. In PEG, the decrement of the independent variable $\vec{v}_{go}$ from previous guidance cycle is not performed in an exact manner, that is, only sensed change in velocity is considered with the start of current iteration cycle as in Eq. (31). Moreover, inaccuracy of the initial guess can induce more iteration and even early divergence of the guidance commands, in other words, guidance instability. Most of the explicit guidance schemes for launch

Please cite this article in press as: Song, E.-J., et al. A comparison of iterative explicit guidance algorithms for space launch vehicles. J. Adv. Space Res. (2014), http://dx.doi.org/10.1016/j.asr.2014.09.025

vehicles including PEG have a general stability problem, especially when the vehicle is close to the injection point. For PEG, the denominators of Eqs. (32) and (33) approach 0 as t_{go} reduces to 0, which explains the intrinsic instability of the scheme. Bittner (1976) and Vittal and Bhat (1993) have tried to solve the problem by formulating a guidance law using a quadratic injection criterion rather than making injection errors to be exactly zero. These previously suggested approaches have focused on eliminating undesirable forms of the guidance parameters equations which make the system unstable; we treated this problem in another way by taking into account the effect of the iterative algorithm for PEG to guidance stability. Although the complete avoidance of the instability is not feasible with the identical equations for the guidance parameters, it can be delayed by selecting initial guess more elaborately. Luenberger and Ye (2008) have described that the solutions of many iterative algorithms for nonlinear problems have sensitivity on the starting points, and we noticed that this kind of effect may become noticeable as the time-to-go approaches 0. If changes in the gravity integrals are additionally considered, Eq. (31) turns into Eq. (44), and this slight modification can correct the inaccuracy caused by the predictor–corrector approach.

$$\vec{v}_{go}(n) = \vec{v}_{go}(n-1) - \Delta\vec{v} - \Delta\vec{v}_{grav} \tag{44}$$

where

$$\Delta\vec{v}_{grav} = \vec{v}_{grav}(n) - \vec{v}_{grav}(n-1) \tag{45}$$

The gravity integral terms can be easily calculated using navigated states information in the current step and the predicted final states obtained from the previous cycle. For example, when using the method proposed by Delporte and Sauvient (1992) and Jaggers (1977), the gravity integral is given by

$$\vec{v}_{grav} = \frac{t_{go}}{2}\left(\vec{g}_f + \vec{g}_o\right) + \frac{t_{go}^2}{12}\left(\dot{\vec{g}}_o - \dot{\vec{g}}_f\right) \tag{46}$$

which is derived from the gravity field approximated by a third-degree polynomial function such as

$$\vec{g} = \sum_{i=0}^{3} a_i t_{go}^i \tag{47}$$

The coefficients a_o, a_1, a_2, and a_3 are evaluated to satisfy the gravity and its first derivative at current position and the predicted final position, $\vec{g}_o, \vec{g}_f, \dot{\vec{g}}_o$, and $\dot{\vec{g}}_f$. More details about this approach are provided in Delporte and Sauvient (1992) and Jaggers (1977). Now $\vec{v}_{grav}(n)$ in Eq. (45) to obtain $\Delta\vec{v}_{grav}$ is evaluated as follows.

$$\begin{aligned}\vec{v}_{grav}(n) = {} & \frac{(t_{go}(n-1) - \Delta T)}{2}\left(\vec{g}_f(n-1) + \vec{g}_o(n)\right) \\ & + \frac{\left(t_{go}(n-1) - \Delta T\right)^2}{12}\left(\dot{\vec{g}}_o(n) - \dot{\vec{g}}_f(n-1)\right)\end{aligned} \tag{48}$$

where ΔT denotes the elapsed time from the previous cycle. Then the convergence characteristics can be improved to a similar level with the linear differential corrector, where ΔT can be considered exactly in the variable t_{go} as shown in Eq. (49).

$$t_{go}(n) = t_{go}(n-1) - \Delta T \tag{49}$$

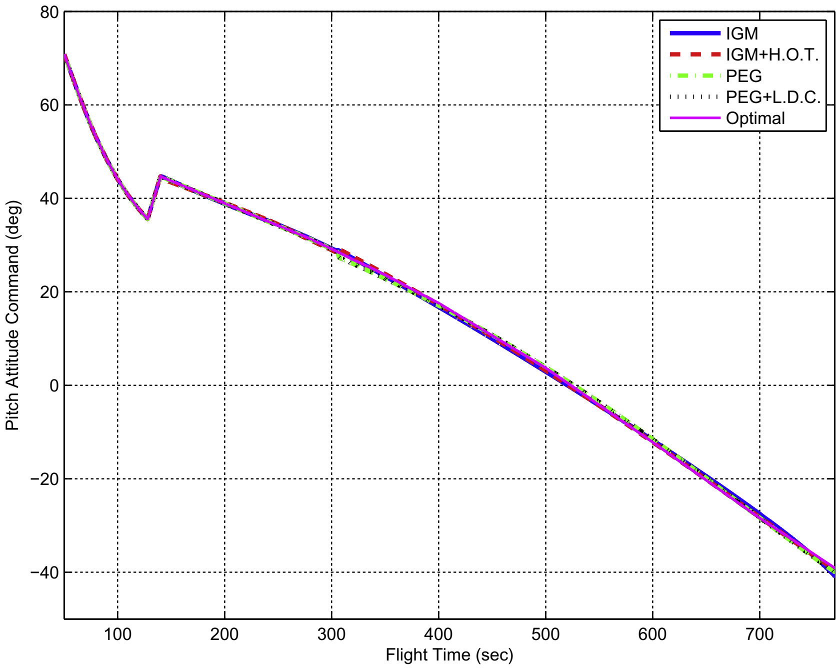

Fig. 1. Pitch attitude command during ascent flight generated by different guidance schemes.

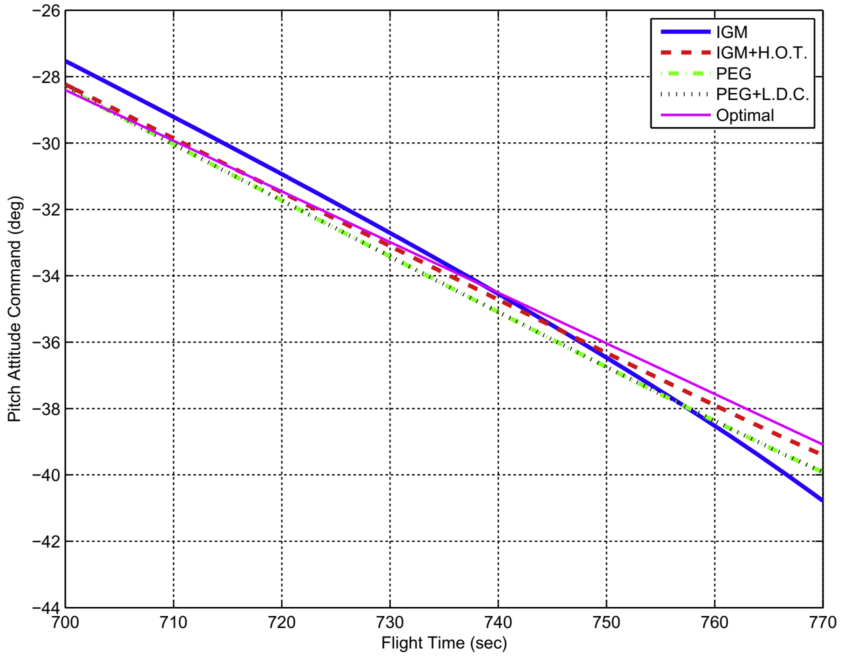

Fig. 2. Pitch attitude command close to the orbit injection point generated by different guidance schemes.

Other guidance parameters $\vec{\lambda}_v$ and $\vec{\dot{\lambda}}$ in Eq. (41) change slowly during ΔT, which means their initial guesses using the results from the previous iteration cycle are so close to the solutions from the current cycle and they do not trigger any convergence problems. Comparison between the original approach and proposed modification is presented in the next section.

Table 1
Nominal trajectory data for the 3-staged launch vehicle model.

	First stage	Second stage	Third stage
Ignition time (sec)	0	128	276
Burn time (sec)	126	144	502

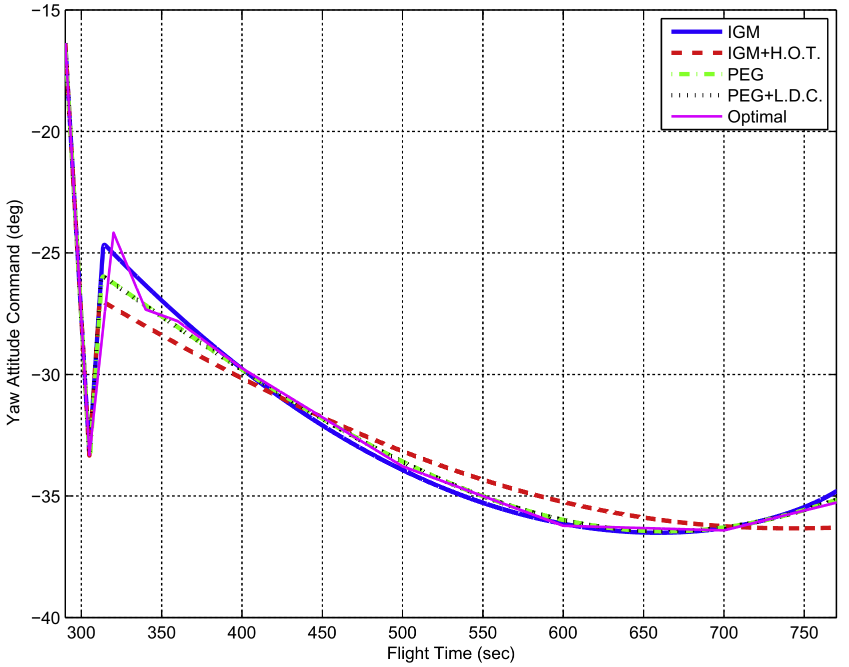

Fig. 3. Yaw attitude command in stage 3 generated by different guidance schemes.

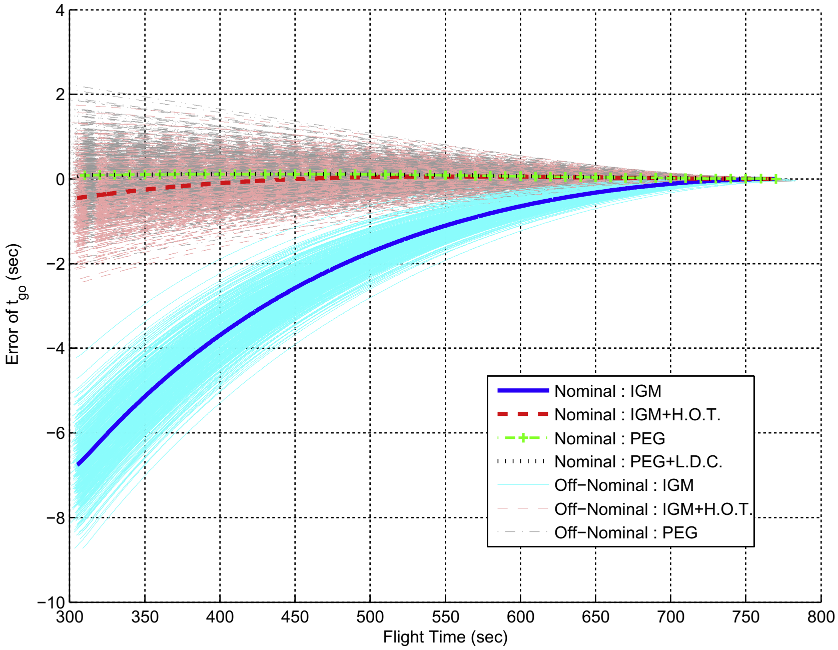

Fig. 4. Prediction error of t_{go} with the nominal and off-nominal conditions. ‘IGM’ shows noticeably larger errors than those of other schemes. ‘IGM + H.O.T.’ fixes the problem and shows similar performance with ‘PEG’.

4. Guidance performance evaluation

We performed a series of 3-DOF (degree of freedom) computer simulations here for an ascent trajectory of a 3-staged launch vehicle model with following five closed-loop guidance algorithms applied during the exoatmospheric phase of the second and third stage flights.

(1) IGM (original)
(2) IGM + H.O.T. (higher-order terms)
(3) PEG (original)
(4) PEG + M.I.G. (modified initial guess)
(5) PEG + L.D.C. (linear differential corrector)

The closed-loop guidance operation is initiated after about 10 (sec) from the ignition of each upper stage. Before then open-loop commands are employed just like for the stage 1 flight, where the vehicle ascends against the dense atmosphere. It is usual to adopt an open-loop logic for this phase for safety and reliability reasons. Target orbits of the stage 2 and stage 3 were set to be their nominal final conditions obtained by trajectory optimization (‘Optimal’ in

Table 2
Stage 3 shutdown sensitivities to the nominal condition with different guidance algorithms.

Guidance algorithm	R(Perigee) (m)	R(Apogee) (m)	Inclination (deg)	Longitude of ascending node (deg)
$v_{go} \leqslant 150$ (m/s)				
IGM	143.08	−306.61	−0.00025	0.00005
IGM + H.O.T.	99.38	−297.31	0.00072	−0.00012
PEG	182.28	−287.85	−0.00010	0.00002
PEG + M.I.G.	182.25	−288.74	−0.00010	0.00002
PEG + L.D.C.	182.27	−287.87	−0.00010	0.00002
$v_{go} \leqslant 500$ (m/s)				
IGM	−1488.84	1347.83	0.00259	−0.00047
IGM + H.O.T.	−100.32	−485.51	0.00162	−0.00028
PEG	−238.50	−392.96	0.00118	−0.00021
PEG + M.I.G.	−238.45	−393.00	0.00118	−0.00021
PEG + L.D.C.	−238.73	−393.02	0.00118	−0.00021
$v_{go} \leqslant 750$ (m/s)				
IGM	−4201.31	3403.95	0.00865	−0.00161
IGM + H.O.T.	−629.79	−347.72	0.00387	−0.00070
PEG	−1228.23	−283.23	0.00580	−0.00106
PEG + M.I.G.	−1228.13	−283.31	0.00119	−0.00021
PEG + L.D.C.	−1228.06	−283.18	0.00580	−0.00106

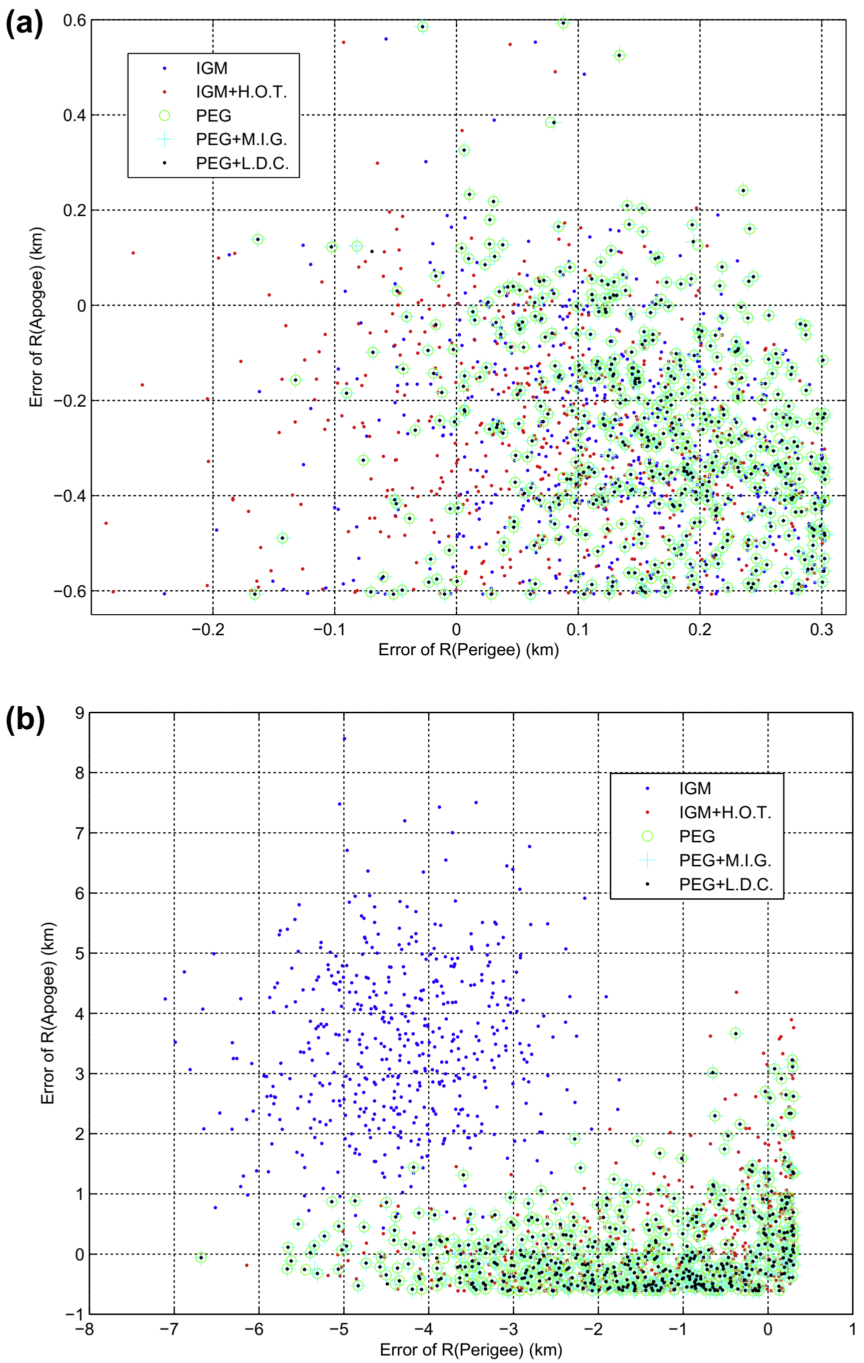

Fig. 5. Scatter plot of insertion perigee error versus apogee error depending on the guidance stop conditions (a) $v_{go} \leqslant 150$(m/s) and (b) $v_{go} \leqslant 750$(m/s). Earlier guidance stop with ‘IGM’ results in larger insertion errors than those of other schemes. ‘PEG’, ‘PEG + M.I.G.’, and ‘PEG + L.D.C.’ provide very similar results.

Figs. 1–3). After the burnout of the third stage engine, the payload is assumed to be injected into the target orbit specified as a 700 km-high near-circular orbit.

To compare above guidance algorithms, here we adopt the same gravity integrals prediction method that Delporte and Sauvient (1992) and Jaggers (1977) have employed in their work. As briefly described in Section 3, the gravity field is modeled with a third-degree polynomial function of time. To ascertain the effect of the higher-order terms described in Section 2, the flight duration of the third stage is set to be much longer than that of the second stage for the nominal trajectory, as shown in Table 1. We evaluated the guidance performance for off-nominal conditions as well as for the nominal condition to validate the suggested methods by conducting the Monte Carlo analysis. The Monte Carlo analysis is a practically useful tool to evaluate the effects of complicated combinations of multiple variables and broadly employed for launch vehicle design and performance analysis (Bell et al., 1993; Hanson and Hall, 2008). For each guidance algorithm,

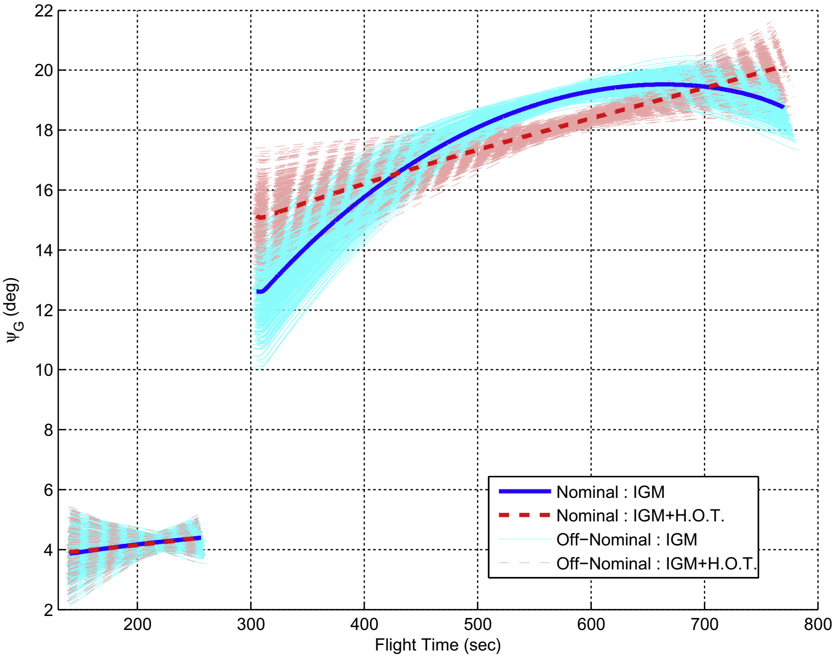

Fig. 6. Yaw attitude command in the guidance frame of IGM with the nominal and off-nominal conditions. The linearity assumption is more valid with 'IGM + H.O.T'.

the Monte Carlo analysis consisting of 500 individual simulations was performed with simultaneously varying error sources modeling the main performance uncertainties in thrust, specific impulse, dry and propellant weights, and so on for each stage. Here each error source was modeled as a random vector with a normal distribution of which the 3σ value was assumed to be about 2 percent.

Figs. 1–3 show the time histories of the attitude commands resolved in a launch inertial frame which is generated by the guidance laws for the simulated flight with the nominal condition. Since the commands generated by 'PEG', 'PEG + M.I.G.', and 'PEG + L.D.C.' are very similar, those generated by 'PEG + M.I.G.' are omitted in Figs. 1–3. It is noticed that the attitude commands generated by different guidance schemes show little difference for the second stage flight with a short duration. However, those generated during the longer, third stage flight show some differences between guidance schemes. For the yaw command, 'IGM' appears to be more close to 'Optimal' than 'IGM + H.O.T', and vice versa for the pitch command. Also as can be expected, 'PEG' and 'PEG + L.D.C.' show no virtual difference and other states omitted here show similar results.

Fig. 4 shows that the time-to-go prediction error of 'IGM' gets noticeably larger than those of other guidance schemes with a longer time-to-go. It also shows that if we consider the higher-order terms, this error can be reduced to a similar level with that of PEG. As previously mentioned, we performed five-hundred different simulations of off-nominal conditions to compare the guidance schemes and to test the effect of inclusion of the higher-order terms to the original IGM. And as depicted in Fig. 4, regardless of the scheme, the off-nominal histories represented by the 500 thin lines show similar characteristics as the nominal profile represented by the bold line.

Orbit injection accuracy for the nominal condition with different guidance stop conditions is shown in Table 2, where the orbit injection errors are defined as the difference between the final injection states obtained from the simulations and the target orbit values used for guidance. The guidance laws are set to be terminated if the velocity-to-go is less than the specified value, and it is noticed that 'IGM' can produce some erroneous engine shutdown time with an early termination of the guidance algorithm, which again may result in orbit injection errors. Attitude commands after guidance termination are usually generated to be constant or to change linearly with time and here we chose the linear commands, so that the attitude rates become constant. Fig. 5 illustrates the targeting results with the off-nominal conditions. As expected, 'IGM' again produces lower performance than others with an earlier guidance stop and 'IGM + H.O.T.' shows similar performance with those of 'PEG's. Fig. 6 shows that the linear angle assumption of IGM in Eqs. (1) and (2) is more valid in 'IGM + H.O.T' during the third stage flight with both the off-nominal and nominal conditions although there are no differences between them in the short flight duration of the stage 2. In Fig. 7 it is observed that the effects of higher-order nonlinear terms increase with remaining flight time for the simulated trajectories.

Table 2 and Fig. 5 also show that 'PEG', 'PEG + M.I.G.', and 'PEG + L.D.C.' have little differences in the orbit injection accuracy. In spite of the similarity in the converged guidance commands illustrated in Figs. 1–3,

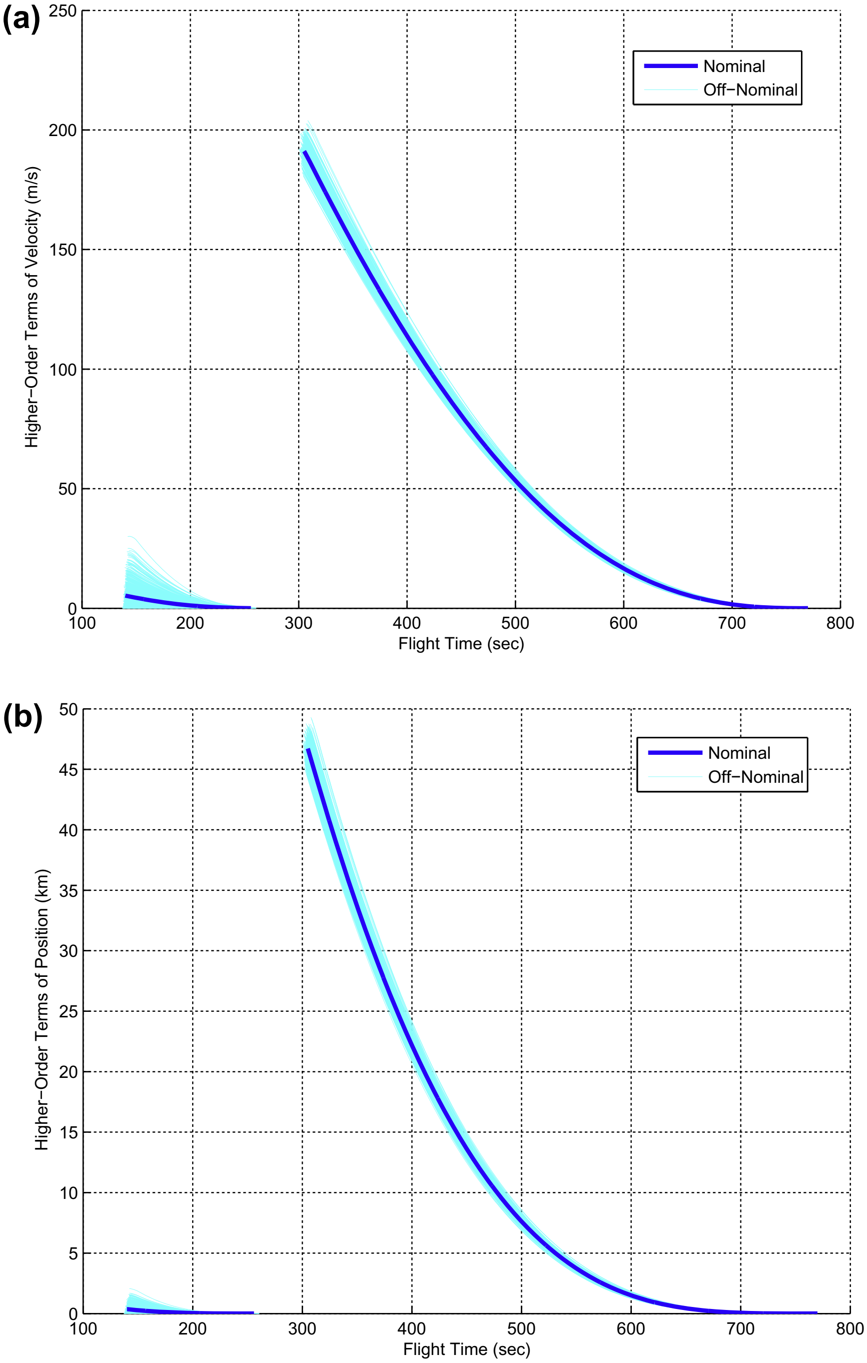

Fig. 7. Higher-order terms of (a) velocity and (b) position in IGM with the nominal and off-nominal conditions.

Fig. 8 illustrating the intermediate values during the iteration process shows that the effect of inaccurate initial guess for the predictor–corrector iteration process of 'PEG' increases as the time-to-go approaches 0. At every start of the iteration, the effect of incorrect guesses appears as the peak points in blue and black lines in Fig. 8. If the time-to-go is not close to 0, after a small number of iterations they converge to similar values with those of 'PEG + M.I.G.' and 'PEG + L.D.C.', which do not have the initial guess effect. However, inaccuracy in the initial guess of $\vec{v}_{go}$ increases as the time-to-go reduces to 0 and produces incorrect guidance parameters different from those of 'PEG + M.I.G.' and 'PEG + L.D.C.'. Fig. 9 explains the reason why the absolute error in the initial guess of v_{go} assumed by the magnitude of $\Delta\vec{v}_{grav}$ in Eq. (45) changes slightly during the flight time, while its relative error increases since v_{go} reduces to 0 as the launch vehicle approaches the target point. The effect of incorrect starting point also increases with the duration of the guidance computation cycle, as expected from Fig. 9. An earlier

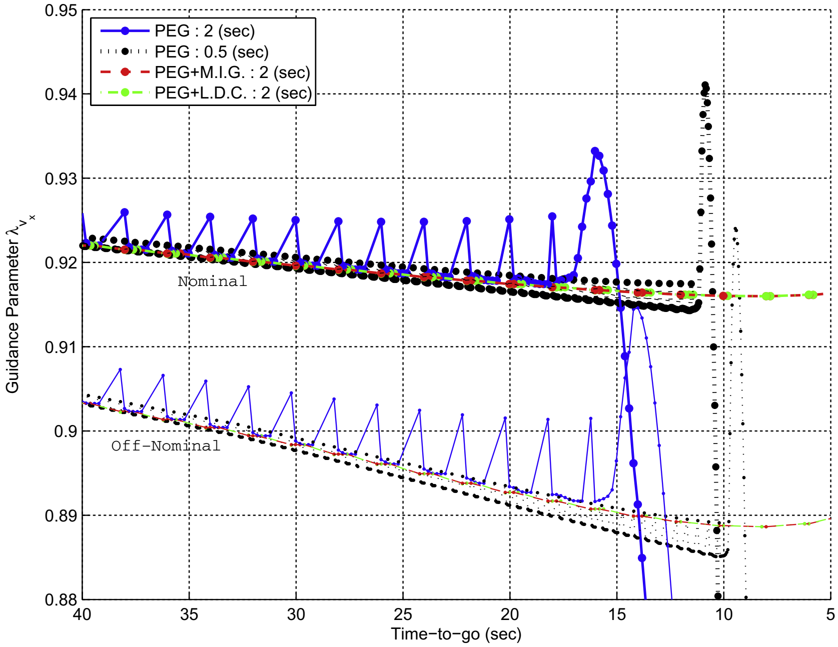

Fig. 8. Change in the guidance parameter of λ_x in PEGs during the iteration process with the nominal and off-nominal conditions. This parameter is sensitive to the initial guess of 'PEG' and the sensitivity also increases with the guidance major cycle update time and as t_{go} approaches 0.

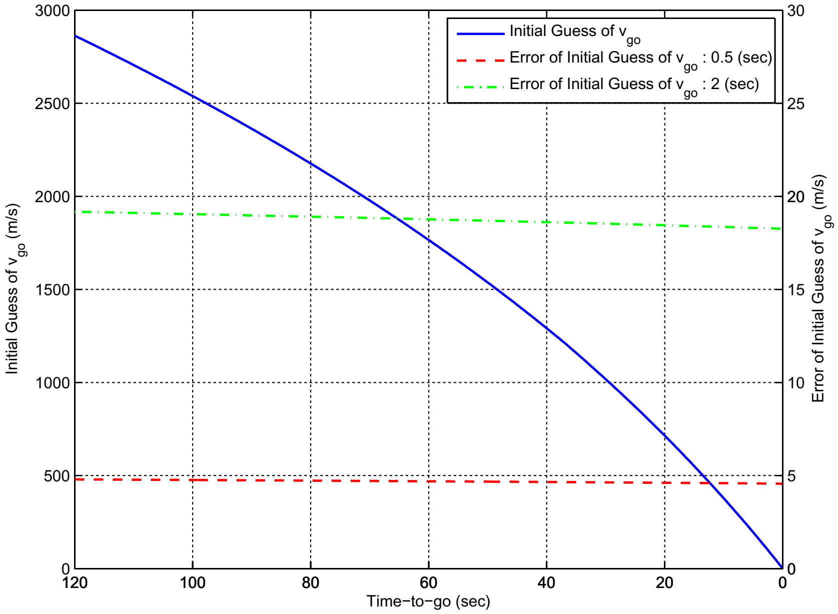

Fig. 9. Initial guess error relative to v_{go}. The relative error increases as t_{go} approaches 0.

divergence is observed in the update cycle of 2.0 (sec) when compared with the 0.5 (sec) case in Fig. 8, and thus an early termination of the guidance is required to prevent the instability. It also affects the number of the iterations to be performed at each guidance cycle, as illustrated in Fig. 10, from which it is observed that 'PEG' requires more iterations than other algorithms do in most cases. Here the iteration is set to be stopped when the difference between the updated and original t_{go} is less than 0.0001 (sec), and this criterion can be mitigated to speed up the computation. Figs. 8 and 10 also include off-nominal results obtained for $+3\sigma$ perturbed thrust conditions of the stage 2, and they also show similar tendencies with the nominal conditions. Also we checked results for the randomly selected 500 perturbed conditions and observed the same tendencies, although detailed results are omitted here.

The simulation results of PEGs imply that although the original predictor–corrector algorithm has an advantage

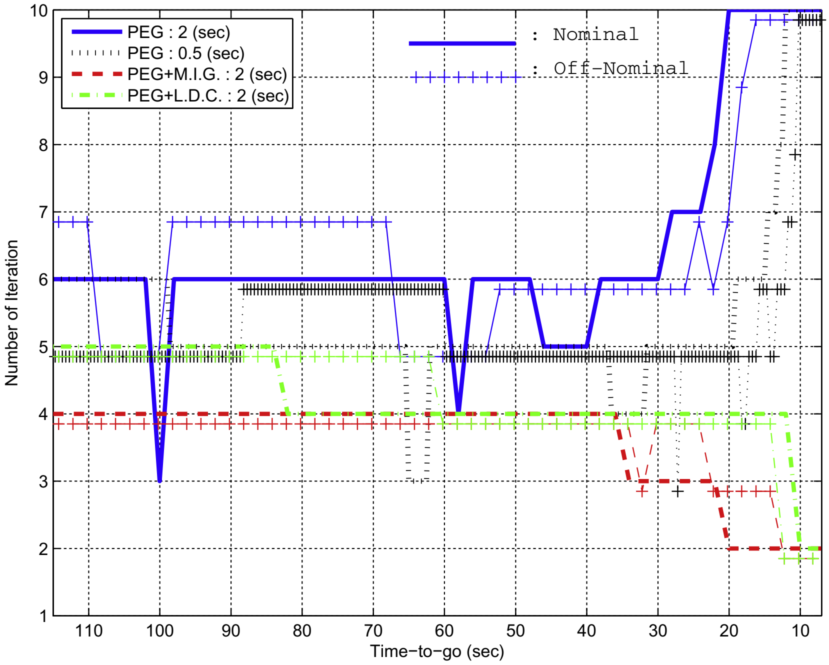

Fig. 10. Number of iterations in PEGs with the nominal and off-nominal conditions. ‘PEG’ with a larger initial guess error results in more iterations than those of others.

having less computational complexity than the differential corrector algorithm does in that it does not require the derivatives of the guidance parameters, the convergence can be often trickier than that of the differential corrector algorithm. It is also shown that a slight modification of the initial guess can improve the convergence characteristics of PEG.

5. Conclusions

This paper considers two renowned explicit guidance algorithms for space launch vehicles, IGM and PEG developed in the space era of 1960–1970s and presents performance comparison and in-depth analysis of each algorithm. We showed that, by including originally ignored higher-order nonlinear terms in IGM, the performance of IGM can be comparable to that of PEG. And the assumption employed in the derivation of IGM that the suboptimal commands are in linear forms becomes more valid with modified IGM. Simulation results for both nominal and off-nominal conditions imply that accurate representation of thrust integrals with higher-order nonlinear terms is necessary, especially for missions with long flight time and/or range. For PEG, we showed that sensitivity to the incorrectly guessed velocity-to-go can be reduced by improving initial guess for the iterative predictor–corrector computation process. Comparison with the linear differential corrector approach is performed to verify the proposed method. In future study, dependence of the convergence characteristics of the iteration process on time-to-go will be analyzed in detail.

References

Bell, S.C., Ginsburg, M.A., Rao, P.P., 1993. Monte Carlo analysis of the titan III/transfer orbit stage guidance system for mars observer mission. AIAA-93-3889.

Bittner, H., 1976. Flat-earth guidance law using in-flight vehicle parameter identification. Automatica 12 (5), 427–443.

Chandler, D.C., Smith, I.E., 1967. Development of the iterative guidance mode with its application to various vehicles and missions. J. Spacecraft Rockets 4 (7), 898–903.

Delporte, M., Sauvient, F., 1992. Explicit guidance law for manned spacecraft. AIAA paper 92-1145.

Hanson, J.M., Hall, C.E., 2008. Learning about ares I from Monte Carlo simulation. AIAA paper 2008-6622.

Hanson, J.M., Shrader, M.W., Cruzen, C.A., 1995. Ascent guidance comparisons. J. Astronaut. Sci. 43 (3), 307–326.

Hardtla, J., Piehler, M., Bradt, J., 1987. Guidance requirement for future launch vehicles. AIAA paper 87-2462.

Jaggers, R.F., 1977. An explicit solution to the exoatmospheric powered flight guidance and trajectory optimization problem for rocket propelled vehicles. AIAA paper 77-1051.

Jessick, M.V., Knobbs, D.L., 1992. Integrated guidance and control algorithms for the national launch system. AIAA-92-4305.

Jutty, K.A., Bhat, M.S., Ghose, D., 2000. Performance of parallel shooting method for closed loop guidance of an optimal launch vehicle trajectory. Optim. Eng. 1 (4), 399–435.

Karacsony, P.J., Cole, C.E., 1970. Application of parameterized guidance. AIAA paper 70-1006.

Luenberger, D.G., Ye, Y., 2008. Linear and nonlinear programming. In: International Series in Operations Research & Management Science, third ed., vol. 116

McHenry, R.L., Brand, T.J., Long, A.D., et al., 1979. Space shuttle ascent guidance, navigation, and control. J. Astronaut. Sci. 27 (1), 1–38.

Schleich, W.T., 1982. The space shuttle ascent guidance and control. AIAA paper 82-1497.

Shrivastava, S.K., Seetharama Bhat, M., Sinha, S.K., 1986. Closed loop guidance of satellite launch vehicle – an overview. J. Inst. Eng. (India) 66 (2), 62–76.

Please cite this article in press as: Song, E.-J., et al. A comparison of iterative explicit guidance algorithms for space launch vehicles. J. Adv. Space Res. (2014), http://dx.doi.org/10.1016/j.asr.2014.09.025

Sinha, S.K., Shrivastava, S.K., 1990. Optimal explicit guidance of multistage launch vehicle along three-dimensional trajectory. J. Guidance 13 (3), 394–403.

Skalecki, L., Martin, M., 1993. General adaptive guidance using nonlinear programming constraint-solving methods. J. Guid. Control Dyn. 16 (3), 517–522.

Song, E.J., Cho, S.B., Park, C.S., et al., 2011. Performance study of an explicit guidance algorithm applicable for upper stages of space launch vehicles. Aerosp. Eng. Technol. 10 (1), 89–97.

Song, E.J., Cho, S.B., Roh, W.R., 2013. Comparison of PEG algorithm with guidance algorithm using linearization approach. In: The 12th Space Launch Vehicle Technology Symposium, pp. 56–62.

Vittal, R.V., Bhat, M.S., 1991. An explicit closed-loop guidance for launch vehicles. Acta Astronaut. 25 (3), 119–129.

Vittal, R.V., Bhat, M.S., 1993. Optimal explicit terminal guidance for a launch vehicle. Acta Astronaut. 29 (4), 249–262.

Please cite this article in press as: Song, E.-J., et al. A comparison of iterative explicit guidance algorithms for space launch vehicles. J. Adv. Space Res. (2014), http://dx.doi.org/10.1016/j.asr.2014.09.025